



US 20120284176A1

(19) **United States**

(12) **Patent Application Publication**
Svendsen et al.

(10) **Pub. No.: US 2012/0284176 A1**

(43) **Pub. Date: Nov. 8, 2012**

(54) **SYSTEMS AND METHODS FOR
COLLABORATIVE ONLINE CONTENT
EDITING**

Publication Classification

(51) **Int. Cl.**
G06F 3/01 (2006.01)
G06Q 50/18 (2012.01)
G06Q 40/00 (2012.01)
(52) **U.S. Cl.** **705/39; 715/753; 705/310**
(57) **ABSTRACT**

(76) Inventors: **Jostein Svendsen**, Saratoga, CA
(US); **Bjørn Rustberggaard**,
Nesøya (NO)

(21) Appl. No.: **13/506,156**

(22) Filed: **Mar. 29, 2012**

Related U.S. Application Data

(60) Provisional application No. 61/468,725, filed on Mar. 29, 2011, provisional application No. 61/564,256, filed on Nov. 28, 2011, provisional application No. 61/564,257, filed on Nov. 28, 2011, provisional application No. 61/564,261, filed on Nov. 28, 2011.

Various embodiments of the invention provide systems and methods for collaborative online content editing, where collaboratively-created content comprising high definition/quality content is created or modified at an online content editing server according to instructions from an online content editor client, and where a proxy version of the resulting collaboratively-created content is provided to online content editor client to facilitate review or further editing of the collaboratively-created content from the online content editor client. In some embodiments, the online content editing server utilizes proxy content during creation and modification operations on the collaboratively-created content, and replaces such proxy content with corresponding higher definition/quality content, possibly when the collaboratively-created content is published for consumption, or when the user has paid for the higher quality content.



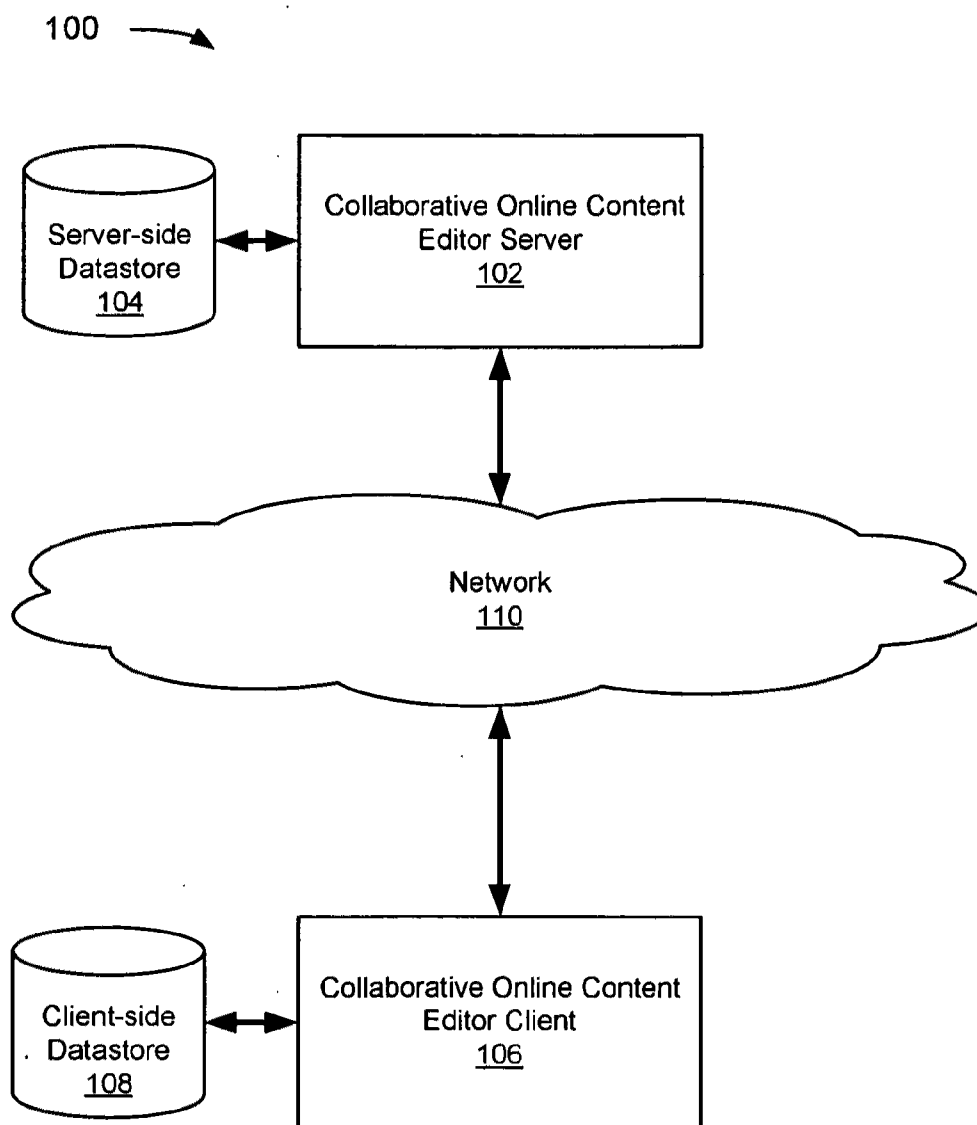


FIG. 1

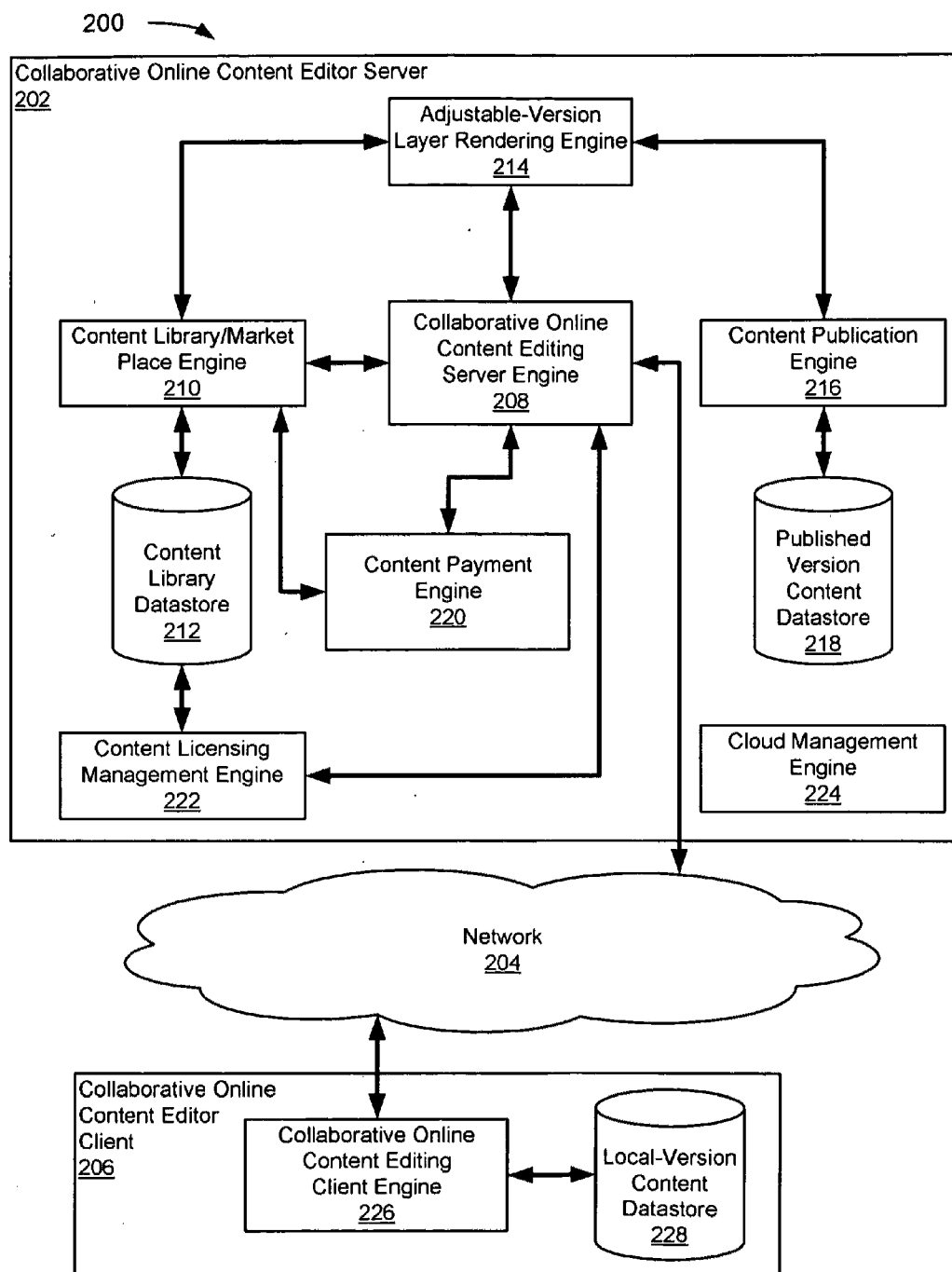


FIG. 2

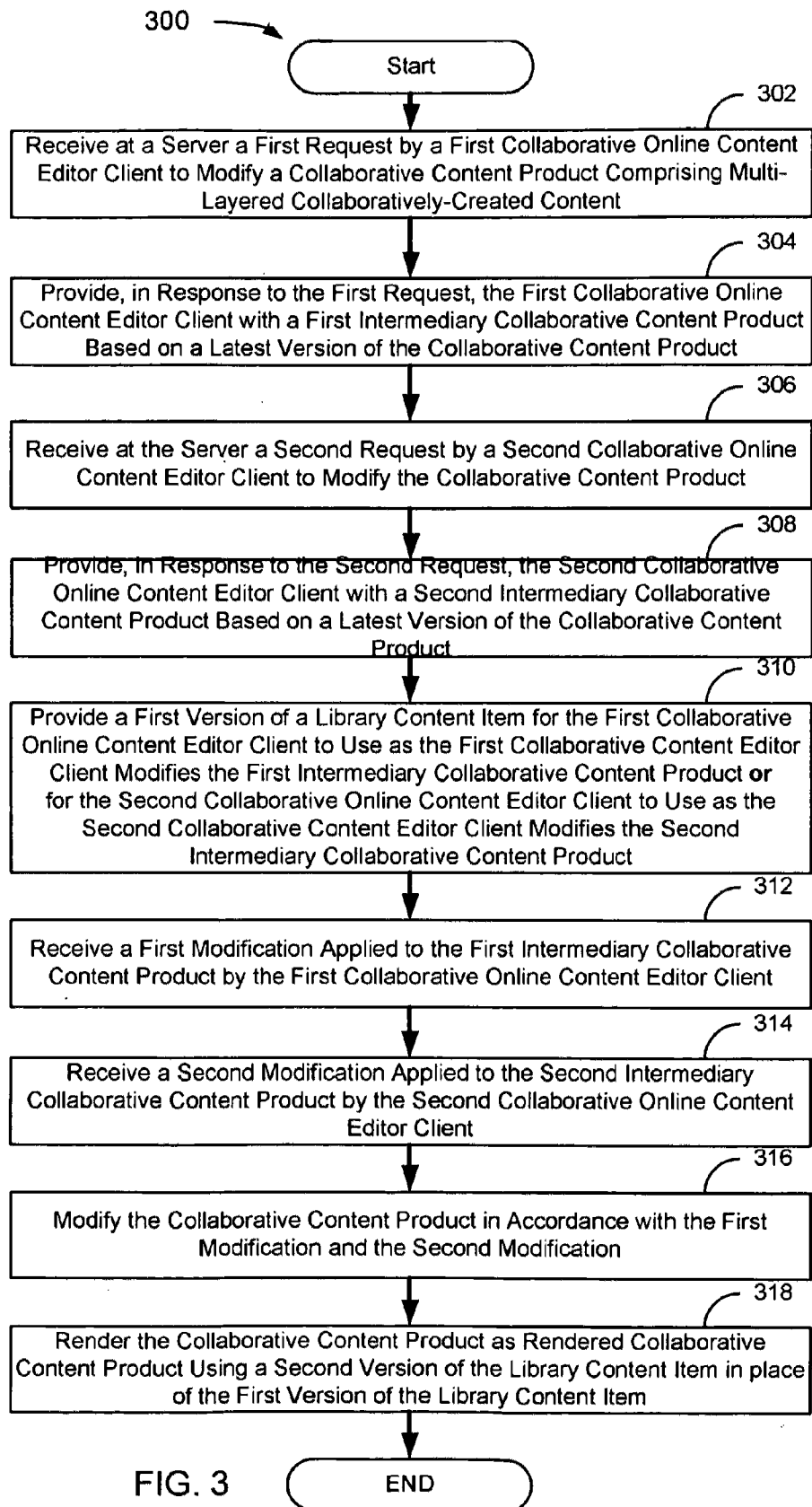


FIG. 3

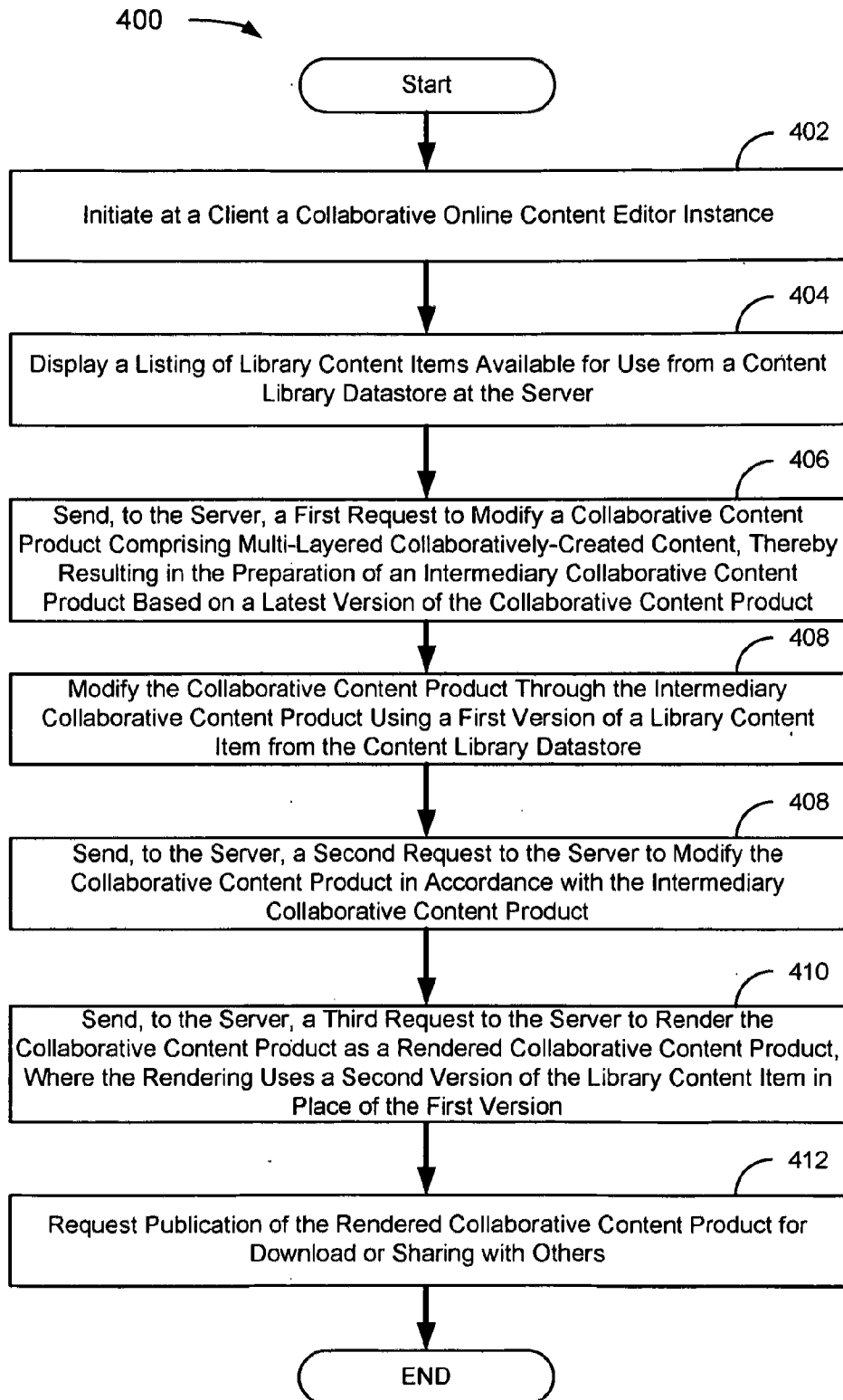


FIG. 4

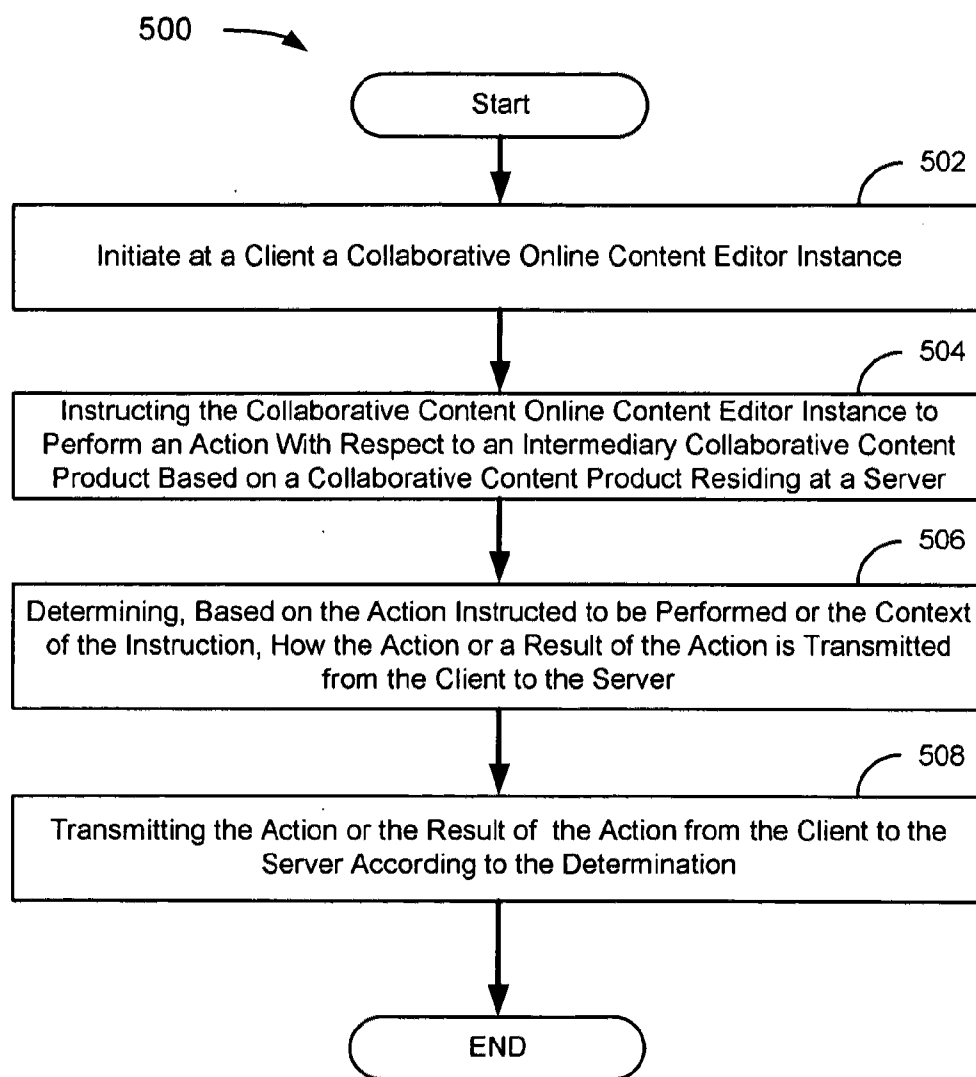


FIG. 5

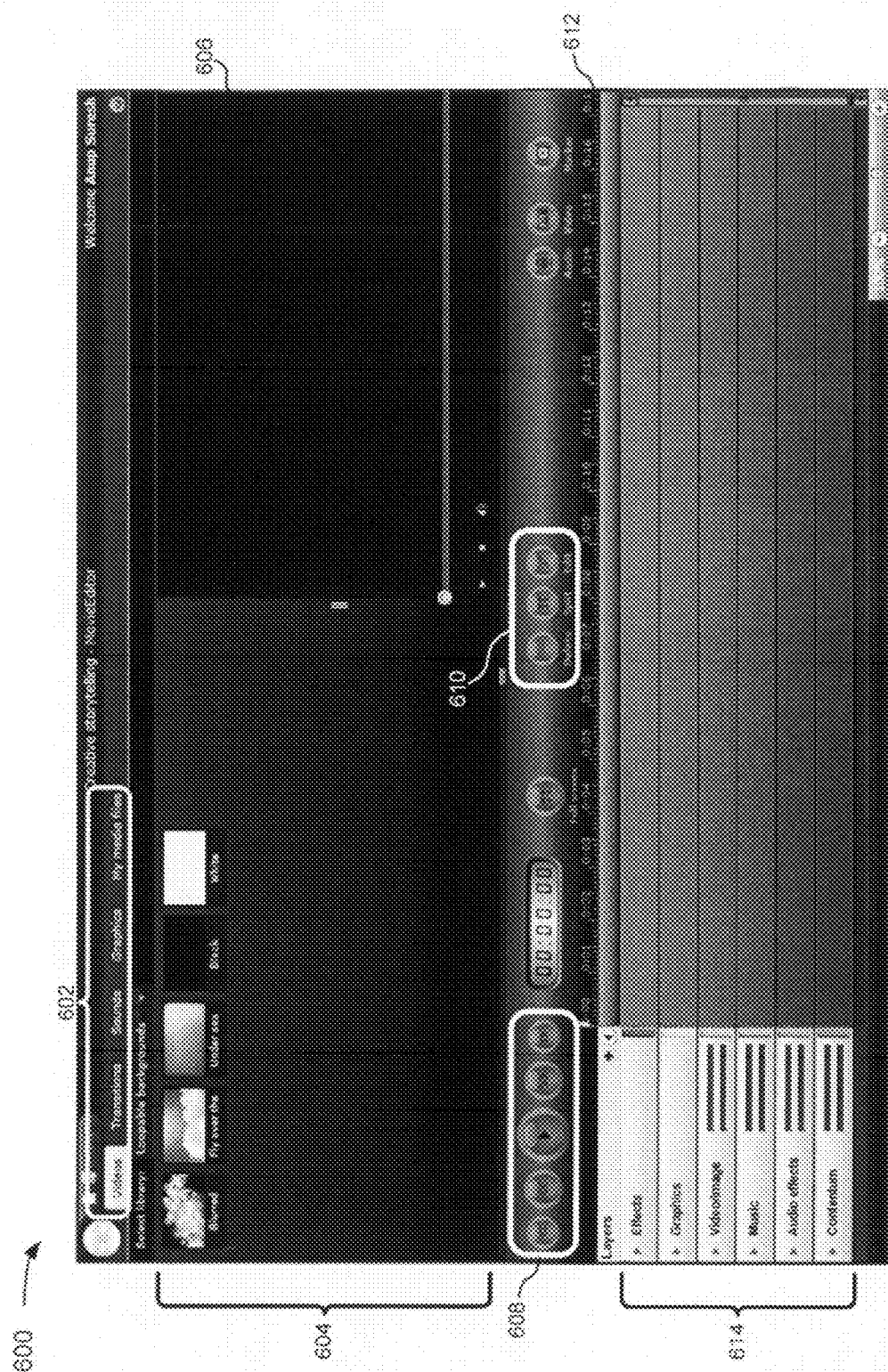


FIG. 6

700 →

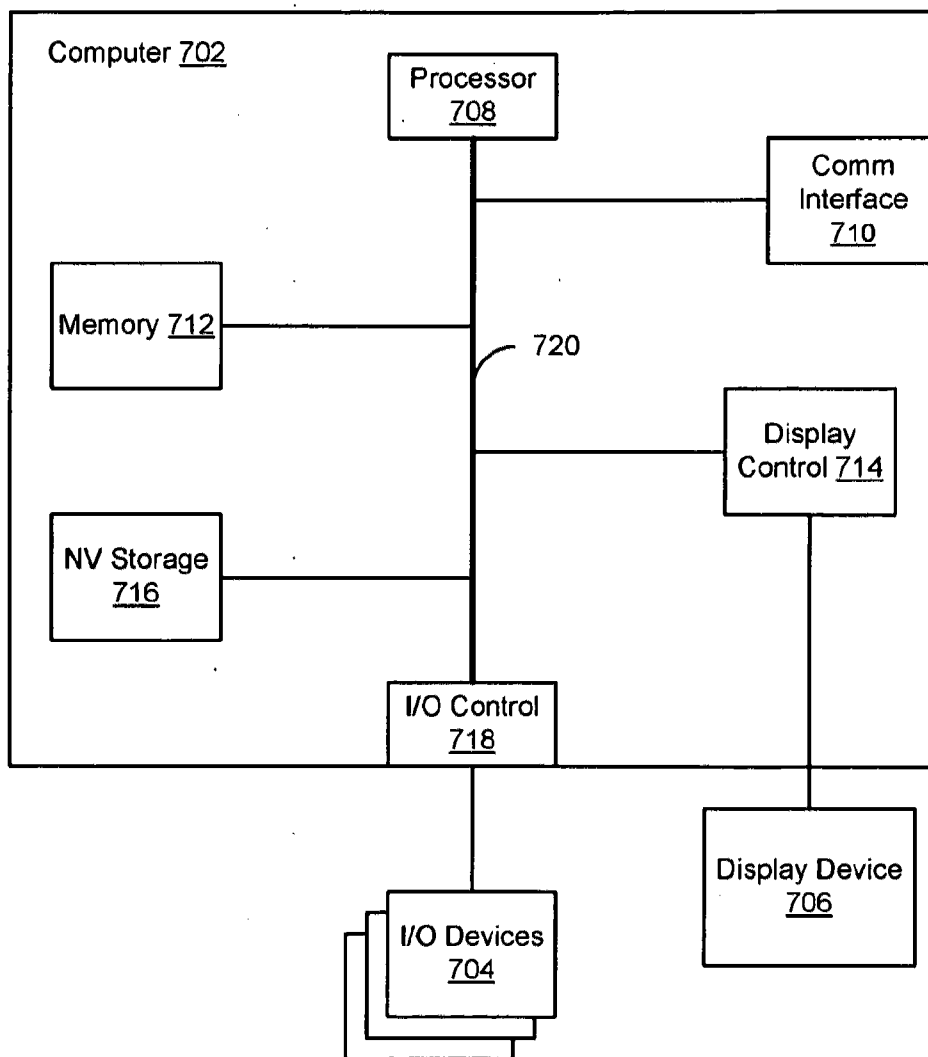


FIG. 7

SYSTEMS AND METHODS FOR COLLABORATIVE ONLINE CONTENT EDITING

PRIORITY CLAIM

[0001] The present application claims benefit of: U.S. Provisional Patent Application Ser. No. 61/468,725 filed Mar. 29, 2011 and entitled “Media Management;” U.S. Provisional Patent Application Ser. No. 61/564,256 filed Nov. 28, 2011 and entitled “Local Timeline Editing for Online Content Editing;” U.S. Provisional Patent Application Ser. No. 61/564,257 filed Nov. 28, 2011 and entitled “Multi-Layer Timeline Content Compilation Systems and Methods;” and U.S. Provisional Patent Application Ser. No. 61/564,261 filed Nov. 28, 2011 and entitled “Systems and Methods for Low Bandwidth Consumption Online Content Editing;” which are incorporated herein by reference.

BACKGROUND

[0002] Generally, audio-video content editing on computer systems involves computationally intensive processes and utilizes larger than usual amounts of computing resources than other computer activities (e.g., word processing, browsing the Internet). This is particularly true when the content editing involves high definition/quality audio or video content, which are well notorious for being large in data size and process heavy during content encoding and decoding processes.

[0003] Unfortunately, due to these computing needs, high definition/quality audio-video content editing has typically been limited to powerful computing systems that tend to have the requisite computing resources to perform content editing quickly. If the same high definition/quality content editing were to be performed on a less powerful computing system, the likely result would be slower or poorer content editing performance, possibly to the point where the content editing becomes impractical or impossible. Consequently, less powerful computing systems, such as older computer systems, netbooks, and particular mobile devices, are either prevented from performing audio-video content editing, or relegated to performing audio-video content editing involving only proxy content.

[0004] Additionally, in certain situations, content editing projects involving collaboration between two or more users is desirable. Unfortunately, if such collaboration were to be facilitated using traditional high definition/quality audio-video content editing tools installed at individual computing systems, each computing system would require sufficient power to operate the tools in a proper and practical manner.

[0005] The foregoing example of trends and issues is intended to be illustrative and not exclusive. Other limitations of the art will become apparent to those of skill in the relevant art upon a reading of the specification and a study of the drawings.

SUMMARY OF VARIOUS EMBODIMENTS

[0006] According to various embodiments, systems and methods for collaborative online content editing are provided, where collaborative content product, comprising collaboratively-created content that can include original version (also referred to as “non-proxy version” or “non-proxy”) content items and/or derivative version (also referred to herein as “proxy version” or simply “proxy”) content items, is

created and/or modified at an collaborative online content editor server according to the modifications submitted by two or more collaborative online content editor clients. Proxy version content items can include, for example, lower quality/definition versions, copyright limited versions, or water-marked versions content items.

[0007] In various embodiments, during collaborative editing of content, the systems or methods can prepare an intermediary collaborative content product (e.g., temporary copy), based on a version (e.g., latest version) of the collaborative content product residing on a collaborative online content editor server and being collaboratively developed by two or more collaborative online content editor clients. The intermediary collaborative content product can be prepared by the server on behalf of each collaborative online content editor client that desires to update or modify the collaborative content product. Once prepared, each collaborative online content editor client can modify the collaborative content product by applying modifications to its respective intermediary collaborative content product, and having the modifications to the intermediary collaborative content product applied to the collaborative content product. Depending on the implementation, the intermediary collaborative content product can be received by a client from a server, updated/modified at the client, and the updates/modifications to the intermediary collaborative content product then sent back to the server for application to the latest version of the collaborative content product residing at the server. Alternatively, the intermediary collaborative content can be stored at the server on behalf of the client, the client instructs the server to perform update/modification operations on the intermediary content product, and server eventually applies the updates/modifications to the intermediary collaborative content product to the latest version of the collaborative content product residing at the server.

[0008] For various implementations, when two or more updates/modifications are to be applied to the collaborative content product, the updates/modification can apply the modification serially in the order in which they are received by the server, based on the priority of the update/modification, or based on the priority of the client submitting the update/modification. Where a conflict exists between two or more updates/modifications to be applied to the collaborative content product (e.g., where two or more modifications are received at or near the same time), if the modifications do not overlap, some embodiments can apply both to the collaborative content product to resolve the conflict. In some embodiments, conflicts between two or more updates/modifications can be resolved by prompting one or more users (e.g., perhaps only those users submitting the updates/modifications) for a resolution.

[0009] Those skilled in the art will appreciate that when a update/modification is applied to a latest version of the collaborative content product, the application will generally result in a new, latest version of the collaborative content product being produced that incorporates the update/modification.

[0010] For some embodiments, the collaborative content product (and, as a result, the intermediary collaborative content product) can be a proxy version or comprise proxy version content items, thereby lowering the use of network bandwidth between a collaborative online content editor client and server and possibly lowering the processing time during operations (e.g., content modification options). When the col-

laborative content product is rendered, the resulting rendered collaborative content product can be considered a non-proxy version of the collaborative content product, where the render collaborative content product comprises non-proxy content items in the place of proxy content items.

[0011] In some embodiments, the collaborative online content editor server or client can utilize proxy content during creation and modification operations on the user-created content, and replaces such proxy content items with corresponding non-proxy content items possibly when the collaborative content product is rendered, when the collaborative content product is published for consumption, or when a user has paid for the use of non-proxy content items.

[0012] By utilizing proxy versions of the user-created content in this manner, systems and methods of some embodiments can utilize less network bandwidth as the online content editor client commands the online content editor server to perform content-related operations on user-created content, and as the online content editor server provides proxy versions of the user-created content for content review and editing purposes. The use of proxy versions of the user-created content also allow the online content editor client to review and edit content without the need for additional computing resources to review user-created content comprising high quality/high definition content.

[0013] In addition to using proxy versions of content for online content editing, various embodiments can dynamically determine what and/or how data should be transferred should be transferred between a collaborative online content editor client and a collaborative online content editor server during collaborative content editing operations. Determination of what and how data should be transferred can depend on the type of editing operation invoked and/or the context in which the operation is invoked.

[0014] For instance, where a collaborative online content editor client applies a modification to its respective intermediary collaborative content product and the intermediary collaborative content product resides at the client, the client can be configured to send the modified intermediary collaborative content product to the server in parts (e.g., chunks or segments) so that the server, for example, can commence applying the modifications as the parts are received by the server (rather than waiting for the modified intermediary collaborative content arriving in its entirety). In another example, to ensure modifications applied one client can be shared contemporaneously with other collaborative clients, the client can be configured to send modifications to the intermediary collaborative content as they are applied locally at the client rather than receiving in a delayed fashion, or the client can be configured to receive from the server updates as updates/modifications from other collaborative clients are applied to the collaborative content product residing on the server.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] FIG. 1 depicts a diagram of an example of a collaborative online content editor system in accordance with some embodiments.

[0016] FIG. 2 depicts a diagram of an example of a client-server system for collaborative online content editing in accordance with some embodiments.

[0017] FIG. 3 depicts a flowchart of an example of a server-side method for collaborative online content editing in accordance with some embodiments.

[0018] FIG. 4 depicts a flowchart of an example of a client-side method for collaborative online content editing in accordance with some embodiments.

[0019] FIG. 5 depicts a flowchart of an example of an optimization method for collaborative online content editing in accordance with some embodiments.

[0020] FIG. 6 depicts an example of client-side user interface for collaborative online content editing in accordance with some embodiments.

[0021] FIG. 7 depicts an example of a system on which techniques described in this paper can be implemented.

DETAILED DESCRIPTION

[0022] The invention can be implemented in numerous ways, including as a process; an apparatus; a system; a composition of matter; a computer program product embodied on a computer readable storage medium; and/or a processor, such as a processor configured to execute instructions stored on and/or provided by a memory coupled to the processor. In this specification, these implementations, or any other form that the invention may take, may be referred to as techniques. In general, the order of the steps of disclosed processes may be altered within the scope of the invention. Unless stated otherwise, a component such as a processor or a memory described as being configured to perform a task may be implemented as a general component that is temporarily configured to perform the task at a given time or a specific component that is manufactured to perform the task. As used herein, the term 'processor' refers to one or more devices, circuits, and/or processing cores configured to process data, such as computer program instructions.

[0023] A detailed description of one or more embodiments of the invention is provided below along with accompanying figures that illustrate the principles of the invention. The invention is described in connection with such embodiments, but the invention is not limited to any embodiment. The scope of the invention is limited only by the claims and the invention encompasses numerous alternatives, modifications and equivalents. Numerous specific details are set forth in the following description in order to provide a thorough understanding of the invention. These details are provided for the purpose of example and the invention may be practiced according to the claims without some or all of these specific details. For the purpose of clarity, technical material that is known in the technical fields related to the invention has not been described in detail so that the invention is not unnecessarily obscured.

[0024] FIG. 1 depicts a diagram of an example of a collaborative online content editor system in accordance with some embodiments. The collaborative online content editor system 100 includes a collaborative online content editor server 102, a server-side datastore 104 coupled to the collaborative online content editor server 102, an collaborative online content editor client 106, a client-side datastore 108 coupled to the collaborative online content editor client 106, and a network 110 through which the online content editor client 106 communicates with the collaborative online content editor server 102. Through this arrangement, the online content editor client 106 can leverage the computing resources and power of the collaborative online content editor server 102 when creating or modifying collaborative content product, especially when the collaborative content product comprises high definition/quality graphics, image, audio, or video content. Often, the collaborative online content editor server 102 com-

prises computing resources that surpass those of the collaborative online content editor client **106**, or computing resources that are better suited for content creation or modification than those of the collaborative online content editor client **106**. Though FIG. 1 depicts a single online content editor client, the system **100** can include multiple online content editor clients that can communicate with the collaborative online content editor server **102**.

[0025] “Collaboratively-created content” as used herein includes any content authored by one or more users using the content creation/editing tools described in this paper, or some other content editing tool. Collaboratively-created content can include content initially based on/started from a vendor provided or user-created content, and then subsequently modified by one or more collaborative users. The collaboratively-created content can be created to be multi-layered content, comprising multiple content layers of different content types. High definition/quality as used herein includes content having definition or quality that is higher than the average definition or quality for the similar content. For example, high definition/quality audio content can include audio clips having a high sampling rate (e.g., 44 KHz), has a higher bit-rate or effective bit-rate (e.g., 256 Kbs), or is encoded in a lossless audio encoding format.

[0026] As used in this paper, an engine includes a dedicated or shared processor and, typically, firmware or software modules that are executed by the processor. Depending upon implementation-specific or other considerations, an engine can be centralized or its functionality distributed. An engine can include special purpose hardware, firmware, or software embodied in a computer-readable medium for execution by the processor. As used in this paper, a computer-readable medium is intended to include all mediums that are statutory (e.g., in the United States, under 35 U.S.C. 101), and to specifically exclude all mediums that are non-statutory in nature to the extent that the exclusion is necessary for a claim that includes the computer-readable medium to be valid. Known statutory computer-readable mediums include hardware (e.g., registers, random access memory (RAM), non-volatile (NV) storage, to name a few), but may or may not be limited to hardware.

[0027] In the example of FIG. 1, the collaborative online content editor system **100** can enable a user at the collaborative online content editor client **106** located remotely on the network **110** to instruct the collaborative online content editor server **102** to create or modify a collaborative content product, containing collaboratively-created content, on behalf of the client **106**. The created or modified collaboratively-created content can include proxy or non-proxy content items, but generally includes only proxy content items while the collaboratively-created content is being created and modified, and then non-proxy content items once the collaboratively-created content is rendered (e.g., for publication or sharing). The collaboratively-created content may be multi-layered content comprising a plurality of content layers, where each content layer comprises one or more content items from a content library, and the content items are high definition/quality content items or versions thereof.

[0028] To collaborate modification operations, on the collaborative content product, between two or more collaborative online content editor clients, the collaborative online content editor server **102** can prepare a copy of the collaborative content product (also referred to herein as an “intermediary collaborative content product”) for each collaborative

online content editor client (e.g., including the collaborative online content editor client **106**) desiring to modify the collaborative content product. For some embodiments, the intermediary collaborative content product can also be prepared as a previous of the latest version of the collaborative content product. Additionally, the intermediary collaborative content product can be a proxy version of the collaborative content product (e.g., to save network bandwidth or reduce processing time). Though the intermediary collaborative content product is generally of the latest version of the collaborative content product, various implementations may perform an intermediary collaborative content product to be based off a previous version of the collaborative content product.

[0029] Once prepared by the collaborative online content editor server **102**, the intermediary collaborative content product can be maintained by and stored at the collaborative online content editor server **102** on behalf of each collaborative online content editor client, can be sent to the respective collaborative online content editor client, or both (e.g., where the intermediary collaborative content product for some collaborative online content editor clients is maintained at the server **102**, and the intermediary collaborative content product for others is maintained at the respective clients). Then, when the collaborative online content editor client **106**, for example, desires to apply a modification to the collaborative content product, it does so using its respective intermediary collaborative content product. In some implementations where the intermediary collaborative content product for the collaborative online content editor client **106** is maintained at the server **102**, the client **106** can instruct the server **102** to perform the desired modifications the respective intermediary collaborative content product and, subsequently, to apply update the latest version of the collaborative content product with the modification to the respective intermediary collaborative content product. In some implementations where the intermediary collaborative content product for the collaborative online content editor client **106** is maintained at the client **106**, the client **106** can directly modify its intermediary collaborative content product and, subsequently, send the modifications applied to its intermediary collaborative content product to the server **102** (which can update the latest version of the collaborative content product with the received modification).

[0030] With respect to some implementations, a modification applied to the collaborative content product (e.g., through use of the intermediary collaborative) by the collaborative online content editor client **106** can include, in addition to content layer modification operations, such operations as: adjusting copyright use limitations on some or all of the collaborative content product, locking some or all portions of the collaborative content product such that some or all of other collaborative online content editor clients are prevented from modifying the designated locked portions, adding watermarks to some or all of the collaborative content product, or tagging objects (e.g., people, places, or things) shown in the collaborative content product.

[0031] As the collaborative online content editor server **102** creates or modifies the collaborative content product, the server **102** can provide the online content editor client **106** with an updated intermediary collaborative content product based on the latest version of the collaborative content product (e.g., after the latest modification from an collaborative online content editor client has been applied). The online content editor client **106** can use the resulting intermediary

collaborative content product (generally comprising having proxy content items) for review or editing purposes as the client continues to modify the collaborative content product.

[0032] As the collaborative online content editor server **102** creates and modifies the collaborative content product (e.g., in accordance with modifications being received from collaborative online content editor clients), one or more versions of the collaborative content product may be stored on the server-side datastore **104**. When the online content editor client **106** receives new or updated versions of intermediary collaborative content product, the client **106** can store these on the client-side datastore **108** before the client **106** directly modifies the new/updated intermediary collaborative content product.

[0033] Those skilled in the art will appreciate that for various embodiments, when a content modification or content update is transferred between the collaborative online content editor server and the collaborative online content editor client **106**, such content modification or update can comprise a list of modification instructions (e.g., including layer identification information, timeline information, content identification information), a copy of the modified content in its entirety, or a copy of the content portions that are modified/updated.

[0034] A datastore can be implemented, for example, as software embodied in a physical computer-readable medium on a general- or specific-purpose machine, in firmware, in hardware, in a combination thereof, or in an applicable known or convenient device or system. Datastores in this paper are intended to include any organization of data, including tables, comma-separated values (CSV) files, traditional databases (e.g., SQL), or other applicable known or convenient organizational formats. Datastore-associated components, such as database interfaces, can be considered “part of” a datastore, part of some other system component, or a combination thereof, though the physical location and other characteristics of datastore-associated components is not critical for an understanding of the techniques described in this paper.

[0035] Datastores can include data structures. As used in this paper, a data structure is associated with a particular way of storing and organizing data in a computer so that it can be used efficiently within a given context. Data structures are generally based on the ability of a computer to fetch and store data at any place in its memory, specified by an address, a bit string that can be itself stored in memory and manipulated by the program. Thus some data structures are based on computing the addresses of data items with arithmetic operations; while other data structures are based on storing addresses of data items within the structure itself. Many data structures use both principles, sometimes combined in non-trivial ways. The implementation of a data structure usually entails writing a set of procedures that create and manipulate instances of that structure.

[0036] Eventually, at the instruction of the online content editor client **106**, the collaborative online content editor server **102** can publish a finalized version of the collaborative content product for download or sharing with others. During publication, the collaborative online content editor server **102** can render the latest version of the collaborative control product (as it resides on the server **102**) such that proxy content items contained in the collaborative control product are replaced with non-proxy content item. For example, during the rendering, the server **102** can replace low definition/quality content items in the collaborative content product with high-quality definition/high quality counterpart or some ver-

sion thereof. For some embodiments, the collaborative online content editor server **102** will replace low definition/quality content with corresponding high-quality definition/high quality content based on payment by a user. Absent payment, the user can be prevented from publishing the collaborative content product, or the user is left with publishing collaborative content product using proxy content items. For example, a user may have to make payment before the collaborative content product is published with non-proxy content items (e.g., high definition/quality content items). The resulting rendered collaborative content product can then be downloaded or shared with others.

[0037] FIG. 2 depicts a diagram of an example of a system for collaborative online content editing in accordance with some embodiments. The system **200** for collaborative online content editing includes a collaborative online content editor server **202**, an online content editor client **206**, and a network **204** facilitating communication between the server **202** and the client **206**. As described herein, the system **200** can enable a user at the online content editor client **206** located remotely on the network **204** to instruct the collaborative online content editor server **202** to create or modify collaborative content product comprising collaboratively-created content possibly including proxy or non-proxy content items. As discussed herein, for some embodiments, the online content editor client **206** can instruct the collaborative online content editor server **202** through a collaborative online content editing client engine **226** located at the client **206**. Subsequently, the collaborative online content editor server **202** can provide, and the online content editor client **206** can receive, a version of the created or modified collaborative content product, such as an intermediary collaborative content product, generally containing proxy content items.

[0038] **100381** In the example of FIG. 2, the collaborative online content editor server **202** comprises a collaborative online content editing server engine **208**, a content library/market place engine **210**, a content library store **212**, an adjustable-version layer rendering engine **214**, a content publication engine **216**, a published version content datastore **218**, and a content payment engine **220**. The collaborative online content editor server **202** further comprises a content licensing management engine **222** and a cloud management engine **224**.

[0039] In the example of FIG. 2, the collaborative online content editing server engine **208** creates or modifies user-created content at the collaborative online content editor server **202** on behalf of the online content editor client **206**. In certain embodiments, the collaborative online content editing server engine **208** can establish a connection with the online content editor client **206** over the network **204**, can receive commands relating to content creation or modification over a network connection, can perform content creation or modification operations in accordance with commands received from the online content editor client **206**, and can transmit to the online content editor client **206** a version of the created or modified collaborative content product (e.g., intermediary collaborative content product). As noted herein, the created or modified user-created content may be multi-layered content comprising one or more content layers, each content layer comprising one or more content items from a content library. The content items utilized can be original content items or proxy versions thereof.

[0040] In order to perform various operations, the collaborative online content editing server engine **208** is coupled to

the content library/market place engine **210**, the adjustable-version layer rendering engine **214**, the content publication engine **216**, and the content payment engine **220**. As discussed in detail below, the collaborative online content editing server engine **208** can access proxy and non-proxy content items and information relating to such content items through the content library/market place engine **210**, can render one or more (content) layers of the collaborative content product at an adjusted quality or version (e.g., copyright-limited use version content items, for-cost content items, free content items, watermarked content items, high-quality content items, or lower quality content items) using the adjustable-version layer rendering engine **214**, and can use the content publication engine **216** to publish the collaborative content product at an adjusted quality or version for download or sharing purposes. The collaborative online content editing server engine **208** can receive pre-payment or post-payment through the content payment engine **220** to permit access to for-purchase content items or to publish the user-defined content items using for-purchase content (e.g., content of varying quality or versions, such as high definition/quality content items).

[0041] In the example of FIG. 2, the content library/market place engine **210** is coupled to the content library datastore **212** and manages the content items stored therein. For some embodiments, the content library/market place engine **210** can be responsible for adding, deleting and modifying content items stored on the content library datastore **212**, for retrieving a listing of content items stored on the content library datastore **212**, for providing details regarding content items stored on the content library datastore **212**, and for providing to other engines content items from the content library. For example, the content library/market place engine **210** can provide content items to the collaborative online content editing server engine **208** as a collaborative user reviews and selects content items to be added to the collaborative content product. In another example, the content library/market place engine **210** can provide content items to the adjustable-version layer rendering engine **214** as the engine **214** renders one or more layers of the collaborative content product.

[0042] The content library/market place engine **210** can also provide a marketplace where user-posted market items, such collaboratively-created content items (e.g., collaborative content product), individual-created, content items, and user uploaded content items, can be marketed to others (e.g., users) having access to the marketplace. The content library/market place engine **210** can be coupled to the content payment engine **220** to facilitate purchases through the marketplace. When a posted market items sells, the content library/market place engine **210** can be configured to credit the user/users associated with the market item with some or all of the funds from the purchase. Depending on the implementation, a user can charge varying prices for vary qualities or versions of a market item. The content library/market place engine **210** can also be configured to allow visitors of the marketplace to rent market items.

[0043] In the example of FIG. 2, the content library datastore **212** stores content items that can be used in the collaborative content product. The content items of the content library datastore **212** can include, without limitation, audio content, video content, image content, graphics contents, and user-provided content. In some embodiments, the user-provided content can include the collaborative content product that was created or modified using the collaborative online

content editor server **202** and subsequently added to the content library datastore **202**, and user content uploaded from the online content editor client **206** to the collaborative online content editor server **202**. The content items of the content library datastore **212** can also include, without limitation, free content that can be used in collaborative content product without need of payment, for-purchase content that can be reviewed only after user payment, content that can be added to collaborative content product only after user payment, and content that can be used in the publication of collaborative content product only after user payment.

[0044] In some embodiments, the high definition/quality content items can be added to, removed from, and adjusted within the user-created item without need of payment. When the user requests publication of the collaborative content product, the collaborative online content editor server **202** can request payment before publication of the collaborative content product is allowed. Alternatively, a payment can be requested in order for the collaborative content product to be published using the high definition/quality content rather a proxy version of the high definition/quality content.

[0045] In some embodiments, the collaborative online content editor server **202** can use proxy versions of the content items when performing creation and modification operations on the collaborative content product (e.g., through the intermediary collaborative content product, where the use of the proxy version continues until publication of the collaborative content product. In some embodiments, the content library/market place engine **210** can provide the server **202** with the proxy version of the content item utilized during creation and modification operations. Depending on the embodiment, the content library/market place engine **210** can generate the proxy version as it is requested by various engines (e.g., the adjustable-version layer rendering engine **214**, or the collaborative online content editing server engine **208**), or the content library/market place engine **210** can provide the proxy version as stored on the content library datastore **212**. As described herein, the proxy version of the high definition/quality content can also be utilized by the collaborative online content editing server engine **208** when transmitting previews of the content to the online content editor client **206** or when transmitting an intermediary collaborative content product to the online content editor client **206**.

[0046] The content stored on the content library datastore **212** (hereafter, also referred to as “content items” or “library content items”) can have various associated properties (i.e., content item properties) that provide details regarding aspects of the content items. In certain embodiments, the content item properties can comprise such information as content type, content duration, content usage permissions (e.g., licensing rights), content cost parameters (e.g., free to use until the collaborative content product is published, or is published using high definition/quality content), content data size, content source (e.g., user provided, or third party author), or content creation or modification date. Through collaborative online content editing server engine **208**, a user can review listings of content items in the content library datastore **212** (e.g., available or unavailable content), review details regarding content items in the content library (e.g., price, usage parameters, content type, duration where applicable, or content definition/quality), manage content items in the content library datastore **212** (e.g., add, delete, or modify content items in the content library), and add content items from the content library datastore **212** to collaborative content product.

[0047] In the example of FIG. 2, the adjustable-version layer rendering engine 214 renders one or more layers of the collaborative content product using content items provided by the content library/market place engine 210 from the content library datastore 212. The content items provided to the adjustable-version layer rendering engine 214 can be an original version of the content item or a lesser definition/quality version of the same. As the adjustable-version layer rendering engine 214 renders layers of the collaborative content product, the collaborative online content editing server engine 208 can specify to the adjustable-version layer rendering engine 214 the desired quality or version for the resulting layers.

[0048] For instance, the collaborative online content editing server engine 208 can specify that the layers of the collaborative content product, comprising high definition/quality content items, be rendered using a low definition/quality version of the high definition/quality content items. Thereafter, the collaborative content product comprising layers of low definition/quality content can be provided to the online content editor client 206 by the collaborative online content editing server engine 208 for preview or editing purposes, as an intermediary collaborative content product, or can be provided to the content publication engine 216 (e.g., for free publication of collaborative content product comprising proxy content items). In some embodiments, the collaborative online content editing server engine 208 can specify that the layers of the collaborative content product, comprising high definition/quality content items, be rendered using the high definition/quality content items, but only at the time of publication through the content publication engine 216, or only after the user submits payment through the content payment engine 220.

[0049] In the example of FIG. 2, the content publication engine 216 receives collaborative content product rendered by the adjustable-version layer rendering engine 214 and publishes the collaborative content product. The collaborative content product can be published such that the collaborative content product can be downloaded and saved by the user or others as a stand-alone content file (e.g., MPEG or AVI file), or such that collaborative content product can be shared to other over the network (e.g., posted to a website so that others can play/view the collaborative content product). Once published, the collaborative content product can be stored on the published version content datastore 218. For some embodiments, the published collaborative content product can be added to the content library datastore 212 for reuse in other collaborative content product. Depending on the embodiment, the published collaborative content product can be added to the content library datastore 212 as for-purchase content (for example, via the content library/market place engine 210, with the sales proceeds being split between amongst the user and the online content editor service provider), or added to the content library datastore 212 as free content available to the public. The user can also define content usage parameters (i.e., licensing rights) for their collaborative content product when the collaborative content product is added to the content library datastore 212.

[0050] In the example of FIG. 2, the content payment engine 220 facilitates user payment to the collaborative online content editor server 202, and can determine the level of functionality provided by the collaborative online content editor server 202, or the level of quality for or the version of content items within the collaborative content product. For

example, once payment has been received by the content payment engine 220 and the content payment engine 220 has informed the collaborative online content editing server engine 208 of such payment, the collaborative online content editing server engine 208 can allow the user to access certain content items in the content library datastore 212 (e.g., for-purchase content), can allow the user to perform certain content creation or modification operations (e.g., splitting a content layer), or can allow the user to publish the collaborative content product with high definition/quality content. In some embodiments, the quality or version of the content item within the collaborative content product can be variable and determined based on the amount payment made by the user. The content payment engine 220 can maintain an account, where the user can maintain a positive balance from which funds are deducted as payments are made to the collaborative online content editor server 202, or where charges are incurred to the account and the user submits payment some time after (e.g., a credit system). In various embodiments, the collaborative online content editing server engine 208 can inform the content payment engine 220 of the costs accrued by the user as he or she uses content items from the content library datastore 212 in the collaborative content product, or as certain functionality of the collaborative online content editing server engine 208 is utilized. As noted herein, the pricing for content items can be stored with the content items in the content library datastore 212.

[0051] In the example of FIG. 2, the content licensing management engine 222 can determine the licensing rights and permissions of content items stored on the content library datastore 212, and inform the collaborative online content editing server engine 208 of such rights and permissions. The collaborative online content editing server engine 208, in turn, can adaptively control its own functionality in accordance with the licensing rights and permissions of the content item as the content item is utilized in the collaborative content product. For instance, where the content licensing rights and permissions of a certain content item restricted the content item from being split during creation or modification of the collaborative content product (i.e., the content item has to remain intact in the collaborative content product), the collaborative online content editing server engine 208 can automatically disable a split content layer command with respect to that certain content item. In another example, the licensing rights and permissions can indicate whether certain content item can be utilized in collaboratively-created content, or has limitations on how many collaborative users can utilize the content item. Other content licensing rights and permissions can include publication limitations on the collaborative content product, or limitations on use of content items based on the existing content of the collaborative content product. The content licensing management engine 222 can inform the collaborative online content editing server engine 208 of the cost of certain content items based on their use in accordance with the licensing rights and permissions. For some embodiments, the authors of the content items can configure the licensing rights and permissions for their content items, which is then stored on the content library datastore 212.

[0052] In some embodiments, where the collaborative online content editor server 202 is implemented using virtual or cloud-based computing resources, such virtual or cloud-based computer resources can be managed through the cloud management engine 224. The cloud management engine 224 can delegate various content-related operations and sub-op-

erations of the server **202** to virtual or cloud-based computer resources, and manage the execution of the operations. Depending on the embodiment, the cloud management engine **224** can facilitate management of the virtual or cloud-based computer resources through an application program interface (API) that provides management access and control to the virtual or cloud-based infrastructure providing the computing resources for the collaborative online content editor server **202**.

[0053] In the example of FIG. 2, the online content editor client **206** comprises a remote online content editor user interface, as part of the collaborative online content editing client engine **226**, and a local-version content datastore **228** coupled to the collaborative online content editing client engine **226**. The collaborative online content editing client engine **226** facilitates the creation or modification of collaborative content product at the collaborative online content editor server **202** by the online content editor client **206**. As noted herein, the collaborative online content editing client engine **226** can establish a connection with the collaborative online content editing server engine **208** over the network **204**, and then issue content creation and modification commands to the collaborative online content editing server engine **208**. In accordance with the issued commands, the collaborative online content editing server engine **208** can perform the content creation or modification operations at the collaborative online content editor server **202**, and can return to the remote online content editor user interface engine **226** a proxy version of the resulting collaborative content product.

[0054] Alternatively, the online content editor client **206** can modify content by receiving an intermediary collaborative content product based on the latest version of the collaborative content product, modifying the intermediary collaborative content product, and then uploading the modifications to the intermediary collaborative content product to the collaborative online content editor server **202** so that the edits can be consistently applied to collaborative content product residing on the server **202**. When the intermediary collaborative content product is uploaded from the online content editor client **206** to the collaborative online content editor server **202** to facilitate modification of the collaborative content product, various implementations can utilize one or more methods for optimizing the network bandwidth usage. More regarding such optimization methods are described herein in association with FIG. 5.

[0055] Depending on the embodiment, the collaborative online content editor server **202** can provide the proxy version of collaborative content product (e.g., intermediary collaborative content product) as a content stream that is buffered and played through the remote online content editor user interface engine **226**, or as a content file that is playable by the remote online content editor user interface engine **226** once the content file has been downloaded to the online content editor client **206** and stored to the local-version content datastore **228**.

[0056] FIG. 3 depicts a flowchart **300** of an example of a server-side method for collaborative online content editing in accordance with some embodiments. In some implementations, the modules of the flowchart **300** and other flowcharts described in this paper are rendered to a permutation of the illustrated order of modules or reorganized for parallel execution. In the example of FIG. 3, the flowchart **300** starts at module **302** with receiving at a server a first request by a first collaborative online content editor client to modify a collabora-

tive content product comprising multi-layered collaboratively-created content. The flowchart **300** continues to module **304** with providing, in response to the first request, the first collaborative online content editor client with a first intermediary collaborative content product based on a latest version of the collaborative content product. When providing the first intermediary collaborative content product, the module **304** may cause the first intermediary collaborative content product to be maintained and stored at the server or, alternatively, cause the first intermediary collaborative content product to be sent to the client where will be maintained and stored.

[0057] In the example of FIG. 3, the flowchart **300** continues to module **306** with receiving at the server a second request by a second collaborative online content editor client to modify the collaborative content product. The flowchart **300** continues to module **308** with providing, in response to the second request, the second collaborative online content editor client with a second intermediary collaborative content product based on a latest version of the collaborative content product. When providing the second intermediary collaborative content product, the module **304** may cause the second intermediary collaborative content product to be maintained and stored at the server or, alternatively, cause the second intermediary collaborative content product to be sent to the client where will be maintained and stored.

[0058] It should be understood that the order in which the server receives the first request and the second request, and the order in which the server provides the first intermediary collaborative content product and the second intermediary collaborative content product can be arbitrary.

[0059] In the example of FIG. 3, the flowchart **300** continues to module **310** with providing a first version of a library content item for the first collaborative online content editor client to use as the first collaborative content editor client modifies the first intermediary collaborative content product or for the second collaborative online content editor client to use as the second collaborative content editor client modifies the second intermediary collaborative content product.

[0060] Depending on the embodiments, the first version can comprise a lower quality version of the library content item while the second version comprises a higher quality version of the library content item, the first version can comprise a derivative version of the library content item while the second version comprises an original version of the library content item, the first version can comprise a watermarked version of the library content item while the second version comprises a non-watermarked version of the library content, and the first version can comprise a first copyright right limited version of the library content item while the second version comprises a second copyright limited version of the library content. As noted herein, the library content item can comprise an image, an audio clip, a video clip, user-provided content, third party-provided content, payment-free content, or for-purchase content.

[0061] In the example of FIG. 3, the flowchart **300** continues to module **312** with receiving a first modification applied to the first intermediary collaborative content product by the first collaborative online content editor client. The flowchart **300** continues to module **314** with receiving a second modification applied to the second intermediary collaborative content product the second collaborative online content editor client. The first and/or second modification (applied to the respective intermediary collaborative content products) may

comprise of a list of modification instructions (e.g., including layer information, content information, and/or timeline information), a set of modified portions of the respective intermediary collaborative content product, or an entire intermediary collaborative content product as modified. As described herein, where a given intermediary collaborative content product resides at the server, receiving a modification applied to the given intermediary collaborative content product can be an internal process of the server where the server obtains the modification from the given intermediary collaborative content product. Where the given intermediary collaborative content product resides at the client, receiving a modification applied to the given intermediary collaborative content product can comprise receiving the modification at the server from the client.

[0062] In the example of FIG. 3, the flowchart 300 continues to module 316 with modifying the collaborative content product in accordance with the first modification and the second modification. Depending on the embodiment, modifying the collaborative content product in accordance with the first modification and the second modification can comprises serially modifying the collaborative content product in accordance with the first modification and then modifying the collaborative content product in accordance with the second modification. This may occur when, for example, the first modification is received before the second modification is received by the collaborative online content editing server engine. This may also occur when, for example, the first modification is associated with a first collaborative user (e.g., administrative user, or content project lead), the second modification is associated with a second collaborative user (e.g., regular user), and the first collaborative user has a higher user credential than the second collaborative user.

[0063] As noted herein, modifying the collaborative content product in accordance with the first modification can comprise entirely replacing the collaborative content product with the first intermediary collaborative content product as modified. Alternatively, modifying the collaborative content product in accordance with the first modification can comprise applying a difference between the collaborative content product and the first intermediary collaborative content product as modified to the collaborative content product. How the modification is performed can depend on user preferences, administrative preferences, or predetermined settings of the collaborative system. It will be well appreciated by those skilled in the art that the collaborative content product can, in some embodiments, be modified in accordance with the second modification in a similar manner.

[0064] In the example of FIG. 3, the flowchart 300 continues to step 318 with rendering the collaborative content product as rendered collaborative content product using a second version of the library content item in place of the first version of the library content item.

[0065] FIG. 4 depicts a flowchart 400 of an example of a client-side method for collaborative online content editing in accordance with some embodiments. In some implementations, the modules of the flowchart 400 and other flowcharts described in this paper are rendered to a permutation of the illustrated order of modules or reorganized for parallel execution. In the example of FIG. 4, the flowchart 400 starts at module 402 with initiating at a client a collaborative online editor instance.

[0066] In the example of FIG. 4, the flowchart 400 continues to module 404 with the collaborative online editor instance displaying a listing of library content items available for use from a content library datastore at a server.

[0067] In the example of FIG. 4, the flowchart 400 continues to module 406 with the collaborative online editor

instance sending, to the server, a first request to modify a collaborative content product comprising multi-layered collaboratively-created content, thereby resulting in the preparation of an intermediary collaborative content product based on a latest version of the collaborative content product. As noted herein, the intermediary collaborative content product once prepared by the server, may reside at the server or be sent to the client for maintenance and storage.

[0068] In the example of FIG. 4, the flowchart 400 continues to module 408 with modifying the collaborative content product through the intermediary collaborative content product using a first version of a library content item from the content library datastore. As described herein, where the intermediary collaborative content product resides at the server, the client can instruct the server to perform the modification operations to the intermediary collaborative content product. Where the intermediary collaborative content product resides at the client, the client can directly perform the modification operations on the intermediary collaborative content product, and then transmit the modifications performed on the intermediary content product to the server.

[0069] Eventually the modification applied to the intermediary content product is consistently applied to the collaborative content product. As such, the flowchart 400 continues to module 410 with sending, to the server, a second request to the server to modify the collaborative content product in accordance with the intermediary collaborative content product.

[0070] In the example of FIG. 4, the flowchart 400 continues to module 412 with sending, to the server, a third request to the server to render the collaborative content product as a rendered collaborative content product, where the rendering uses a second version of the library content item in place of the first version.

[0071] In the example of FIG. 4, the flowchart 400 continues to module 414 with requesting publication of the rendered collaborative content product for download or sharing with others.

[0072] FIG. 5 depicts a flowchart of an example of an optimization method for collaborative online content editing in accordance with some embodiments. Use of the optimization method can, for some embodiments, allow for efficient use of network bandwidth between an online low-bandwidth consumption client and an online low-bandwidth consumption server and/or allow for less processing time during content modification operations. Depending on the embodiment, the optimization method can be implemented at a collaborative online content editor client, at a collaborative online content editor server, or both. Once implemented, the optimization method can determine how and when data transfer between the collaborative online content editor client and the collaborative online content editor server can be performed in a manner which reduces the amount of data being transferred. Generally, the optimization method makes the transfer determination based on the type of action instructed to be performed, based on the context in which action is instructed to be performed, preferences set at the online low-bandwidth consumption client or the online low-bandwidth consumption server, or some combination thereof.

[0073] Those skilled in the art will appreciate that depending on the implementation, the optimization method can be implemented for content transfer from the online low-bandwidth consumption client to the collaborative online content editor server, can be implemented for content transfer from the collaborative online content editor server to the collaborative online content editor client, or both. Additionally, though the optimization methods described herein are primarily described in association with content transfers, those skilled in the art will appreciate that the optimization methods

can be utilized for other information being transferred between collaborative online content editor client and the collaborative online content editor server including, for example, timeline information. Furthermore, in some implementations, the modules of the flowchart 500 and other flowcharts described in this paper are reordered to a permutation of the illustrated order of modules or reorganized for parallel execution.

[0074] In the example of FIG. 5, the flowchart 500 starts at module 502 with initiating at a client a collaborative online editor instance. Once initiated, a user at the client can issue commands, such as content creation or modification commands relating to the collaborative content product, through the collaborative online content editor instance to the server. In response, the collaborative online content editor instance can instruct the server to perform operations on the collaborative content product residing at the server via an intermediary collaborative content product, which may be residing at the server or residing at the client. As noted herein, the client can be implemented by any known or convenient computing device including, for example, mobile computing devices, netbooks, and desktop. Generally, the server possesses computing resources in excess of those of the client, or the server possesses computing resources better suited for content creation or modification than those of the client.

[0075] In the example of FIG. 5, the flowchart 500 continues to module 504, instructing the collaborative content online content editor instance to perform an action with respect to the intermediary collaborative content product, which is based on the collaborative content product. Examples of actions can include modification of a layer in the intermediary collaborative content product, changing attributes of the intermediary collaborative content product, adding, removing content, or adjusting content items in the intermediary collaborative content product, or updating the collaborative content product with the current version of the intermediary collaborative content product.

[0076] The context in which the instruction to perform the action is made, or the action to be performed can have an influence on how the method optimizes data transfers between the client and server. Examples of context aspects that can be considered include whether the instruction is requested in a collaborative setting (e.g., where two or more collaborative online content editor clients are modifying a single collaborative content product), content editor user preferences (e.g., early start on rendering processes), and whether one or more other instructions have been requested concurrently with the instruction requested at module 504.

[0077] In the example of FIG. 5, the flowchart 500 continues to module 506 with determining based on the action instructed to be performed or the context of the instruction, how the action or a result of the action is transmitted from the client to the server. In one example, where a content editor user indicates a preference (e.g., through user settings) for server processes (e.g., rendering or conversion) to be performed on content as early as possible (e.g., as content data begins to arrive at the server from the client), the module 506 can determine that any upload of intermediary collaborative content product from the client to the server should be transferred in parts (e.g., content data chunks or segments), thereby enabling the server to commence processing of the content as the parts are received (rather than waiting for the intermediary collaborative content product in its entirety). By transferring data from the client to the server in parts, the client can cause the server to pause or stop content processing mid-performance (e.g., collaborative content product update process when a mistake is discovered in the modified intermediary collaborative content product), thereby potentially saving time and resources.

[0078] In yet another example, two or more collaborative online content editor users are collaboratively performing content edits to the same collaborative content product, where each content editor user is respectively editing his or her respective intermediary collaborative content product residing locally at the content editor user's client. In such an instance, the module 506 can determine that portions of the intermediary collaborative content product residing at the respective clients are to be uploaded to the server as the portions are modified by the individual content editor users. In doing so, various implementation can try to ensure that the latest changes applied by individual collaborative users to the user-created content can be shared amongst other collaborative users.

[0079] In the example of FIG. 5, the flowchart 500 continues to step 508 with transmitting the action or the result of the action from the client to the server according to the determination.

[0080] FIG. 6 depicts an example of client-side user interface for collaborative online content editing in accordance with some embodiments. With respect to some implementations, the client-side user interface of FIG. 6 can control the content creation and editing operations performed on a collaborative content being developed collaboratively by two or more users. In particular, the client-side user interface can control a collaborative online content editing client engine operating at a client, a collaborative online content editing server engine operation at a server, or both to facilitate the creation and editing operations on the collaborative content. As described herein, for various implementations, the client-side user interface can cause various engines to operate such that an intermediary collaborative content product is prepared by the server and received by a client from the server, the intermediary collaborative content product is updated/modified at the client, and the updated/modified intermediary collaborative content product is sent back to the server (e.g., in order to update the latest version of the collaborative content product residing at the server). Additionally or alternatively, the client-side user interface can cause various engines to operate such that an intermediary collaborative content prepared and stored at a server on behalf of the client, the client instructs the server to perform update/modification operations on the intermediary content product, and the client instructs the server to update/modify the collaborative content product (e.g., possibly the latest version of the collaborative content product) using the updated/modified intermediary collaborative content product at the server. The behavior and/or results of the client-side user interface based on user input can be based on individual user preferences, administrative preferences, predetermined settings, or some combination thereof.

[0081] The client-side user interface can be configured such that a user can request performance of creation and modification operations on the collaborative content product being collaboratively developed. The client-side user interface can be transferred from a server to a client as a module that can then be operated on the client. For example, the client-side user interface can comprise a client-side applet or script that is downloaded to the client from the server and then operated at the client (e.g., through a web browser). Additionally, the client-side user interface can operate through a plug-in that is installed in a web browser. User input to the client-side user interface can cause a command relating to online content editing, such as a content layer edit command or a content player/viewer command, to be performed at the client or to be transmitted from the client to the server.

[0082] The client-side user interface 600 includes multiple controls and other features that enable a user at a client to control the creation or modification of collaborative content

products. In the example of FIG. 6, the client-side user interface 600 includes a tabbed menu bar 602, a content listing 604, a content player/viewer 606, content player/viewer controls 608, content layer editor controls 610, a content timeline indicator 612, and a content layering interface 614.

[0083] In the example of FIG. 6, the client-side user interface 600 includes the tabbed menu bar 602 that allows the user to select between different content types (e.g., video, audio, or images) as they search for content available to them from a content library. Here, the tabbed menu bar 602 allows the user to select between reviewing “Videos,” “Sounds,” “Graphics” (e.g., text or images), or personal (media) content (i.e., “My media files”) that is available from the content library. The personal content can be that which the user uploaded to their account on the server, that which the user already created on the server, or both. Those of ordinary skill in the art would appreciate that in some embodiments, the tabbed menu bar 602 can include additional content types (e.g., “3D Video”) from which the user can select as they search for content available to them from the content library.

[0084] The tabbed menu bar 602 also enables the user to select “Transitions,” which can be predefined (e.g., vendor provided), user-created, or collaboratively-created content transitions inserted between two content items in a layer of collaboratively-created content. For instance, with respect to video content (i.e., video clips), available transitions can include a left-to-right video transition which once inserted between a first video clip and a second video clip, cause the first video clip transition to the second video clip in a left-to-right manner. Likewise, with respect to audio content (i.e., audio clips), available transitions can include a right-to-left transition which once inserted between a first audio clip and a second audio clip, causes the first audio clip to fade into to the second audio clip starting from the right audio channel and ending at the left audio channel.

[0085] As the user selects between the content types in the tabbed menu bar 602, a listing of available content matching the selected content type can be displayed to the user accordingly. In the example of FIG. 6, the content listing 604 displays a list of content available from the content library. For some embodiments, the content listing 604 can list the available content with a thumbnail image configured to provide the user with a preview of the content. For example, for video content, the thumbnail image may be a moving image that provided a brief preview of the video content. With respect to image content, the thumbnail preview may be a smaller sized version (i.e., lower resolution version) of the image content. In certain embodiments, a content item listed in content listing 606 can be further previewed in a content player/viewer 606, configured to play audio or video or display larger resolution images. The content listing 604 can also provide details regarding the listed content including, without limitation, a source of the content, a date of creation for the content, a data size of the content, a time duration of the content (where applicable), licensing information relating to the content, and cost of using the content.

[0086] In the example of FIG. 6, a user can utilize the player/viewer 606 to preview content items from the content library listed in the content listing 604. The content player/viewer 606 can also provide a preview of collaboratively-created content being created through the client-side user interface 600. In one example, the user can create collaboratively-created content, containing one or more video and audio content items from the content library, and then preview that collaboratively-created content through the content player/viewer 606. Depending on the embodiment, the content being previewed can be from a latest version of the collaborative content product residing at the server, a rendered version of the collaborative content product residing at

the server, a latest version of the intermediary collaborative content product residing at the server for the client, or a latest version of the intermediary collaborative content product locally residing at the client. Where content being played or shown is provided from the server, such content can be streamed from the server to the client as such the content is played or shown through the content player/viewer 606. In some embodiments, where content being played or shown is provided from the server, such content can be first downloaded to the client before it is played or shown through the content player/viewer 606.

[0087] As described herein, the content library items or the collaborative content product shown through the content player/viewer 606 can be of a proxy version rather than the actual version of the content residing on the server or the content that is eventually rendered by the server. Because the proxy content requires less bandwidth when being transferred from the server to a client, some embodiments can lower the data bandwidth consumption between the client and server as content is created, edited, rendered or previewed through the client-side user interface 600, and can possibly decrease processing time at the client and/or the server during creation, modification, rendering, or preview operations.

[0088] In the example of FIG. 6, a user controls the operations of the content player/viewer 606 using the content player/viewer controls 608. The content player/viewer controls 608 can include control commands common to various players, such as previous track, next track, fast-backward, fast-forward, play, pause, and stop. For some embodiments, a user input to the content player/viewer controls 608 can result in a content player/viewer command instruction being transmitted from the client to the server.

[0089] In the example of FIG. 6, the content layer editor control 610 comprises controls that enable the user to edit content layers of collaborative content product. Through the content layer editor control 610, a user can implement edits to a content layer of the collaborative content product residing on the server. The content layer editor control 610 can include edit controls that enable a user to add, delete or modify one or more content layers of collaborative content product. Example edit controls include, without limitation, adding a content layer, deleting a content layer, splitting a single content layer into two or more content layers, and editing properties of a content layer.

[0090] In the example of FIG. 6, the content timeline indicator 612 visually assists a user in determining a temporal position of a content layer in collaborative content product or a content item in the content layer. For instance, the content timeline indicator 612 can comprise a time marker that indicates a temporal start point or a temporal end point for a content layer or a content item in the content layer. In some embodiments, the length of the content timeline indicator 612 can adapt according to the overall duration of the collaboratively-created creation, or can be adjusted according to a user-setting.

[0091] In the example of FIG. 6, the content layering interface 614 enables a user to access and modify content layers of the collaborative content product. The content layering interface 614 can comprise a stack of content layer slots, where each content layer slot graphically presents all the content layers of a particular content type associated to the collaborative content product. Example content types include, without limitation, graphical content (e.g., “Graphics”), video content (e.g., “Video”), image content (e.g., “Image”), and audio content (e.g., “Audio effects”). The content layering interface 614 can also comprise image, video or audio effects, which can be applied to the various content types.

[0092] In some embodiments, the user can add content to a new or existing content layer of the collaborative content

product by “dragging-and-dropping” content items from the content listing 604 into the content layering interface 614. Further, in various embodiments, the user can graphically modify a temporal position or duration of a content layer or a content item within the content layer. For instance, the user can drag-and-drop the graphically represented start or end of a content item to adjust the duration of the content item (thereby the temporal start of temporal end of the content item) in the collaborative content product.

[0093] FIG. 7 depicts an example of a system on which techniques described in this paper can be implemented. The computer system 700 may be a conventional computer system that can be used as a client computer system, such as a wireless client or a workstation, or a server computer system. The computer system 700 includes a computer 702, I/O devices 704, and a display device 706. The computer 702 includes a processor 708, a communications interface 710, memory 712, display controller 714, non-volatile storage 716, and I/O controller 718. The computer 702 may be coupled to or include the I/O devices 704 and display device 706.

[0094] The computer 702 interfaces to external systems through the communications interface 710, which may include a modem or network interface. It will be appreciated that the communications interface 710 can be considered to be part of the computer system 700 or a part of the computer 702. The communications interface 710 can be an analog modem, ISDN modem, cable modem, token ring interface, satellite transmission interface (e.g. “direct PC”), or other interfaces for coupling a computer system to other computer systems.

[0095] The processor 708 may be, for example, a conventional microprocessor such as an Intel Pentium microprocessor or Motorola power PC microprocessor. The memory 712 is coupled to the processor 708 by a bus 770. The memory 712 can be Dynamic Random Access Memory (DRAM) and can also include Static RAM (SRAM). The bus 770 couples the processor 708 to the memory 712, also to the non-volatile storage 716, to the display controller 714, and to the I/O controller 718.

[0096] The I/O devices 704 can include a keyboard, disk drives, printers, a scanner, and other input and output devices, including a mouse or other pointing device. The display controller 714 may control in the conventional manner a display on the display device 706, which can be, for example, a cathode ray tube (CRT) or liquid crystal display (LCD). The display controller 714 and the I/O controller 718 can be implemented with conventional well known technology.

[0097] The non-volatile storage 716 is often a magnetic hard disk, an optical disk, or another form of storage for large amounts of data. Some of this data is often written, by a direct memory access process, into memory 712 during execution of software in the computer 702. One of skill in the art will immediately recognize that the terms “machine-readable medium” or “computer-readable medium” includes any type of storage device that is accessible by the processor 708 and also encompasses a carrier wave that encodes a data signal.

[0098] The computer system 700 is one example of many possible computer systems which have different architectures. For example, personal computers based on an Intel microprocessor often have multiple buses, one of which can be an I/O bus for the peripherals and one that directly connects the processor 708 and the memory 712 (often referred to as a memory bus). The buses are connected together through bridge components that perform any necessary translation due to differing bus protocols.

[0099] Network computers are another type of computer system that can be used in conjunction with the teachings provided herein. Network computers do not usually include a

hard disk or other mass storage, and the executable programs are loaded from a network connection into the memory 712 for execution by the processor 708. A Web TV system, which is known in the art, is also considered to be a computer system, but it may lack some of the features shown in FIG. 7, such as certain input or output devices. A typical computer system will usually include at least a processor, memory, and a bus coupling the memory to the processor.

[0100] Some portions of the detailed description are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of operations leading to a desired result. The operations are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

[0101] It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the following discussion, it is appreciated that throughout the description, discussions utilizing terms such as “processing” or “computing” or “calculating” or “determining” or “displaying” or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system’s registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

[0102] Techniques described in this paper relate to apparatus for performing the operations. The apparatus can be specially constructed for the required purposes, or it can comprise a general purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a computer readable storage medium, such as, but is not limited to, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, magnetic or optical cards, any type of disk including floppy disks, optical disks, CD-ROMs, and magnetic-optical disks, or any type of media suitable for storing electronic instructions, and each coupled to a computer system bus.

[0103] Although the foregoing embodiments have been described in some detail for purposes of clarity of understanding, the invention is not necessarily limited to the details provided.

We claim:

1. A system for collaborative online content editing, the system comprising:

- a collaborative online content editing server engine;
- an adjustable-version layer rendering engine coupled to the collaborative online content editing server engine;
- a content library engine coupled to the collaborative online content editing server engine;
- a content library datastore coupled to the content library engine, wherein the content library datastore comprises a library content item;

an adjustable-version layer rendering engine;
 wherein, in operation:
 the collaborative online content editing server engine modifies multi-layered collaboratively-created content in accordance with a first modified version of the multi-layered collaboratively-created content and in accordance with a second modified version of the multi-layered collaboratively-created content, wherein the first modified version results from a first collaborative online content editing client engine modifying a first copy of the multi-layered collaboratively-created content and the second modified version results from a second collaborative online content editing client engine modifying a second copy of the user created content;
 the content library engine provides a first version of the library content item for the first collaborative online content editing client engine as the first collaborative online content editing client engine modifies the first copy using the first version of the library content item, or provides the first version of the library content item for the second collaborative online content editing client engine as the second collaborative online content editing client engine modifies the second copy using the first version of the library content item;
 the adjustable-version layer rendering engine renders the multi-layered collaboratively-created content using a second version of the library content item in place of the first version;
 the content library engine provides the second version of the library content item to the adjustable-version layer rendering engine as the adjustable-version layer rendering engine renders the multi-layered collaboratively-created content using the second version.

2. The system of claim 1, wherein the first version comprises a lower quality version of the library content item, and the second version comprises a higher quality version of the library content item.

3. The system of claim 1, wherein the second version comprises an original version of the library content item.

4. The system of claim 1, wherein the first version comprises a watermarked version of the library content item and the second version comprises a non-watermarked version of the library content.

5. The system of claim 1, wherein the first version comprises a first copyright right limited version of the library content item and the second version comprises a second copyright limited version of the library content.

6. The system of claim 1, wherein the library content item comprises an image, an audio clip, or a video clip.

7. The system of claim 1, wherein the library content item comprises user-provided content or third party-provided content.

8. The system of claim 1, wherein the library content item comprises payment-free content or for-purchase content.

9. The system of claim 1, wherein the content library engine comprises a derivative version content generation engine;
 wherein, in operation:
 the derivative version content generation engine generates the first version from an original version of the library content item, and provides the first version to the first collaborative online content editing client engine or the second collaborative online content editing client engine; or

the derivative version content generation engine generates the second version from the original version of the library content item, and provides the second version to the adjustable-version layer rendering engine.

10. The system of claim 1, wherein, in operation:

the adjustable-version layer rendering engine renders the multi-layered collaboratively-created content using the second version based on a user preference, a payment from a user, a system preference, or a system condition.

11. The system of claim 10, further comprising a content payment engine coupled to the collaborative online content editing server engine, wherein the collaborative online content editing server engine comprises:

a payment processing engine;
 a content version determination engine;

wherein, in operation:

the payment processing engine processes the payment;
 the content version determination engine determines the second version based on a condition selected from a group consisting of an amount of the payment or a success in processing the payment.

12. The system of claim 1, further comprising a content publication engine coupled to the collaborative online content editing server engine;

wherein, in operation:

the content publication engine publishes the multi-layered content for consumption after the multi-layered content has been rendered by the adjustable-version layer rendering engine.

13. The system of claim 1, further comprising a content licensing management engine coupled to the collaborative online content editing server engine;

wherein, in operation:

the content licensing management engine obtains a licensing parameter for the second version, wherein the licensing parameter comprises a use limitation associated with the second version;

the content licensing management engine provides the licensing parameter to the collaborative online content editing server engine to limit use of the second version according to the use limitation.

14. The system of claim 1, further comprising a content licensing management engine coupled to a content payment engine;

wherein, in operation:

the content licensing management engine obtains a licensing parameter for the second version, wherein the licensing parameter comprises a cost associated with the second version;

the content licensing management engine provides the licensing parameter to the content payment engine to charge for use of the second version according to the cost.

15. The system of claim 1, wherein the collaborative online content editing server engine modifies the multi-layered collaboratively-created content in accordance with the first modified version and the second modified version by serially modifying the multi-layered collaboratively-created content in accordance with the first modified version and then modifying the multi-layered collaboratively-created content in accordance with the second modified version.

16. The system of claim 15, wherein the first modified version is received by the collaborative online content editing

server engine before the second modified version is received by the collaborative online content editing server engine.

17. The system of claim 15, wherein the first modified version is associated with a first collaborative user, the second modified version is associated with a second collaborative user, and the first collaborative user has a higher credential than the second collaborative user.

18. The system of claim 1, wherein the collaborative online content editing server engine modifies the multi-layered collaboratively-created content in accordance with the first modified version by replacing the multi-layered collaboratively-created content with the first modified version.

19. The system of claim 1, wherein the collaborative online content editing server engine modifies the multi-layered collaboratively-created content in accordance with the first modified version by applying a difference between the multi-layered collaboratively-created content and the first modified version to the multi-layered collaboratively-created content.

20. The system of claim 1, wherein the first collaborative online content editor client modifying the first copy of the multi-layered collaboratively-created content involves the collaborative online content editing server engine modifying the first copy based on a modification instruction received from the first collaborative online content editing client engine.

21. The system of claim 1, wherein the first collaborative online content editor client modifying the first copy of the multi-layered collaboratively-created content involves the first collaborative online content editing client engine directly modifying the first copy.

22. A method for collaborative online content editing, the method comprising:

receiving, at a collaborative online content editing server, a first request by a first collaborative online content editor client to modify a collaborative content product comprising multi-layered collaboratively-created content;

providing, in response to the first request, the first collaborative online content editor client with a first intermediary collaborative content product based on the collaborative content product, wherein the first intermediary collaborative content product comprises a first copy of the multi-layered collaboratively-created content as the multi-layered collaboratively-created content exists in the collaborative content product when the first intermediary collaborative content product is generated;

receiving, at the collaborative online content editing server, a second request by the second collaborative online content editor client to modify the collaborative content product;

providing, in response to the second request, the second collaborative online content editor client with a second intermediary collaborative content product based on the collaborative content product, wherein the second intermediary collaborative content product comprises a second copy of the multi-layered collaboratively-created content as the multi-layered collaboratively-created content exists in the collaborative content product when the second intermediary collaborative content product is generated;

providing a first version of a library content item for the first collaborative online content editor client as the first collaborative online content editor client modifies the first intermediary collaborative content product using the first version of the library content item, or for the

second collaborative online content editor client as the second collaborative online content editor client modifies the second intermediary collaborative content product using the first version of the library content item; receiving a first modification applied to the first intermediary collaborative content product by the first collaborative online content editor client; receiving a second modification applied to the second intermediary collaborative content product by the second collaborative online content editor client; modifying the work content product in accordance with the first modification and the second modification; rendering the collaborative content product as a rendered collaborative content product, wherein the rendering uses a second version of the library content item in place of the first version.

23. The method of claim 22, wherein the first version comprises a lower quality version of the library content item, and the second version comprises a higher quality version of the library content item.

24. The method of claim 22, wherein the second version comprises a first copyright right limited version of the library content item.

25. The method of claim 22, wherein the first version comprises a watermarked version of the library content item and the second version comprises a non-watermarked version of the library content.

26. The method of claim 22, wherein the first version comprises a first copyright right limited version of the library content item and the second version comprises a second copyright right limited version of the library content.

27. The method of claim 22, wherein the library content item comprises an image, an audio clip, or a video clip.

28. The method of claim 22, wherein the library content item comprises user-provided content or third party-provided content.

29. The method of claim 22, wherein the library content item comprises payment-free content or for-purchase content.

30. The method of claim 22, further comprising publishing the rendered collaborative content product for consumption.

31. The method of claim 22, wherein the first intermediary collaborative content product resides at the collaborative online content editing server, and the method further comprises:

receiving, at the collaborative online content editing server, a modification instruction from the first collaborative online content editor client;

modifying the first intermediary collaborative content product according to the modification instruction.

32. The method of claim 22, further comprising generating the first version of the library content item before providing the first version of the library content item.

33. The method of claim 22, wherein rendering the multi-layered collaboratively-created content using the second version is based on a user preference, a payment from a user, a system preference, or a system condition.

34. The method of claim 33, further comprising:

processing the payment from a user;

determining the second version based on an amount of the payment or based on success in processing the payment.

35. The method of claim 22, further comprising:

obtaining a licensing parameter for the second version, wherein the licensing parameter comprises a use limitation associated with the second version;

limiting use of the second version according to the use limitation.

36. The method of claim 22, further comprising:
obtaining a licensing parameter for the second version,
wherein the licensing parameter comprises a cost associated with the second version;
charging for use of the second version according to the cost.

37. The method of claim 22, wherein modifying the collaborative content product in accordance with the first modification and the second modification comprises serially modifying the collaborative content product in accordance with the first modification and then modifying the collaborative content product in accordance with the second modification.

38. The method of claim 37, wherein the first modification is received before the second modification is received by the collaborative online content editing server engine.

39. The method of claim 37, wherein the first modification is associated with a first collaborative user, the second modification is associated with a second collaborative user, and the first collaborative user has a higher user credential than the second collaborative user.

40. The method of claim 22, wherein modifying the collaborative content product in accordance with the first modification comprises replacing the collaborative content product with the first intermediary collaborative content product as modified.

41. The method of claim 22, wherein modifying the collaborative content product in accordance with the first modification comprises applying a difference between the collaborative content product and the first intermediary collaborative content product as modified to the collaborative content product.

42. A system for collaborative online content editing, the system comprising:

a means for receiving, at a collaborative online content editing server, a first request by a first collaborative online content editor client to modify a collaborative content product comprising multi-layered collaboratively-created content;

a means for providing, in response to the first request, the first collaborative online content editor client with a first intermediary collaborative content product based on the

collaborative content product, wherein the first intermediary collaborative content product comprises a first copy of the multi-layered collaboratively-created content as the multi-layered collaboratively-created content exists in the collaborative content product when the first intermediary collaborative content product is generated;

a means for receiving a second request by the second collaborative online content editor client to modify the collaborative content product;

providing, in response to the second request, the second collaborative online content editor client with a second intermediary collaborative content product based on the collaborative content product, wherein the second intermediary collaborative content product comprises a second copy of the multi-layered collaboratively-created content as the multi-layered collaboratively-created content exists in the collaborative content product when the second intermediary collaborative content product is generated;

a means for providing a first version of a library content item for the first collaborative online content editor client as the first collaborative online content editor client modifies the first intermediary collaborative content product using the first version of the library content item, or providing the first version of the library content item for the second collaborative online content editor client as the second collaborative online content editor client modifies the second intermediary collaborative content product using the first version of the library content item;

a means for receiving a first modification applied to the first intermediary collaborative content product by the first collaborative online content editor client;

a means for receiving a second modification applied to the second intermediary collaborative content product by the second collaborative online content editor client;

a means for modifying the work content product in accordance with the first modification and the second modification;

a means for rendering the collaborative content product as a rendered collaborative content product, wherein the rendering uses a second version of the library content item in place of the first version.

* * * * *